

In the patent application, claims 1, 3-7, 9-13 and 15 are pending. In the office action, all pending claims are rejected.

At page 2 of the office action, claim 1, 3-7, 9-13 and 15 are rejected under 35 U.S.C. 103(a) as being unpatentable over *Chew et al.* (U.S. Patent No. 6,664,991, hereafter referred to as *Chew*) in view of *Clark et al.* (U.S. Patent No. 5,995,101, hereafter referred to as *Clark*).

In rejecting claims 1, 7 and 13, the Examiner states that *Chew* discloses an electronic device capable of carrying out a series of operational steps as claimed, except that *Chew* fails to disclose a command symbolized by an icon. In particular, the Examiner states that *Chew* discloses carrying out a command (input function) and providing a message associated with the command (context menu or tool tip). The Examiner points to *Clark* for disclosing tool tips that are commonly associated with icons and that when a user selects the icon with a pointing device, the command associated with the icon is carried out (col.1, lines 10-14). Applicant respectfully disagrees.

First, the tool tip as disclosed in *Clark* is not compatible with the list of entries 300 or 700 as disclosed in *Chew*. Tool tip is used to provide one-direction information, such as a short textual label that appears over an icon to provide a brief identification of the program function associated with the icon. In contrast, the list 300 or 700 of the entries or options is editable by a user.

Second, even when the tool tip as disclosed in *Clark* is used in combination with the tap-and-hold gesture methodology as disclosed in *Chew*, the combination does not render the claimed invention obvious.

It is respectfully submitted that the invention as claimed in claim 1 includes the steps of:

- 1) contacting the screen at a designated area by the physical object; and
- 2) removing the physical object from the screen before a selected time has expired to cause the electronic device to carry out the command, or
- 3) keeping the physical object at the designated area longer than the selected time to cause the electronic device to provide the message, and

4) removing the physical object from the screen after step 3 to cause the electronic device to carry out the command, or

5) moving the physical object off the designated area while keeping the physical object on the screen after step 3 to end the message, wherein the command is symbolized by an icon and the message is associated with the command, and wherein the icon is displayed at the designated area.

According to claim 1, after the physical object contacts the screen at the designated area, one of three events occurs:

- (I) A message associated with the command is provided if the contact is longer than a predetermined time duration (see step 3 above);
- (II) The command is carried out if the contact is shorter than the predetermined time duration (step 2) or the physical object is removed from the screen after the message is displayed (step 4); and
- (III) The message is ended if the physical object is moved out of the designated area while keeping the physical object on the screen (step 5).

Regardless whether *Clark* discloses an icon as claimed, *Chew* does not disclose the steps necessary to bring about the three events as listed above.

In rejecting claims 1, 7 and 13, the Examiner points to col. 3, lines 33-67 to show that *Chew* discloses an electronic device capable of carrying out a command and providing a message associated with the command. The Examiner points to col. 1, line 56 – col.2, line 9 to show that *Chew* discloses providing the message if the contact is longer than a predetermined time duration. The Examiner further points to col. 7, lines 1-8 to show that *Chew* discloses that the user may remove the physical object before the selected time has expired, so that the context menu is not displayed and the “stylus up event is dispatched to the application, and that tapping may cause selection of an entry.

It is respectfully submitted that, at col. 3, lines 33-56, *Chew* only discloses how the electronic device 68 is constructed. At col.3, lines 57-67, *Chew* provides the general description of the invention as follows:

The present invention provides a means for identifying gestures made by the user that are intended to cause a hand-held pen-based device to display a context menu. In response to an identified gesture, the present invention notifies a parent application that the gesture occurred, thereby allowing the parent application to display a context menu if it wants to. If the parent application does not wish to display a context menu, the gesture is ignored and normal processing continues. In some embodiments, the parent application substitutes a tool tip or pop-up help for the context menu.

In the above paragraph, *Chew* discloses that a user may perform a gesture and the device displays a context menu, if it wants to, in response to the gesture. The “gesture” as described in the above paragraph is a context-menu gesture performed by the user by a stylus input (col. 1, lines 61 – 63). The context menu gesture can be made with a tap-and-hold gesture (col. 4, lines 14-16). When these gestures are made, a context-menu appears (col. 1, lines 63-65). In one embodiment, the context menu appears if the stylus is kept on the designated area longer than a specified time duration (col. 1, line 66 – col.2, line 3). In another embodiment, double tapping the stylus is considered a context menu gesture. The double-tapping or double-click gesture is described at col.6, lines 9-16. However, this double-tapping feature is irrelevant to the claimed invention.

At col. 1, line 56 to col.2, line 9, *Chew* discloses:

The present invention provides a user interface for a pen-based device that allows users to activate the full "card" for an entry using a single tap while also giving the user the ability to reach a menu of context dependent commands using a simple gesture. In particular, the present invention recognizes certain stylus and/or button inputs as context menu gestures. When these gestures are made, a context menu appears. The user is then able to select a command in the context menu that is to be performed.

Under one embodiment, the gesture consists of pressing and holding the stylus on a selection for a system-specified length of time. If the user does not move the stylus more than a system specified distance from the original location for that time, a context

menu gesture is recognized and the application displays a context menu. In other embodiments, pressing and holding an external button on the pen-based device while an entry is highlighted is considered a context menu gesture. In still further embodiments, double tapping the stylus or pressing and holding an external button while tapping the stylus are considered context menu gestures.

In the above paragraphs, *Chew* discloses how a context menu can be caused to display by a tap-and-hold gesture and what the user can do after the context menu is displayed. However, what the user can do with the context menu is irrelevant to the claimed invention.

At col. 7, lines 1-8, *Chew* discloses:

The user may also lift the stylus before the context menu gesture is complete. This results in a stylus up event message, which is not within the parameters for a tap-and-hold gesture at step 1206. As such, the stylus up event means that the user is not performing a tap-and-hold. Since a tap-and-hold is not being performed, the event is dispatched to the parent application at step 1208 and the method ends at step 1212.

In the above paragraph, *Chew* discloses the result of lifting the stylus before the specified time duration expires. In particular, as shown in Figure 12, when the tap-and-hold gesture within the gesture parameter is not detected at step 1206, the input events will no longer be captured at step 1210 and the gesture recognition method is terminated at step 1212.

Furthermore, *Chew* discloses that if the stylus has moved outside the contact area, the user is probably trying to perform a drag operation and not a context menu gesture. In such a case, the stylus move event is dispatched to the application at step 1208 for stopping the capturing event messages at step 1210 (col. 6, lines 51-59).

In the above paragraph, *Chew* discloses that the drag operation is equivalent to the stylus up event before the specified time duration expires.

In sum, according to the tap-and-hold gesture method as disclosed in *Chew*, a user may be able to reach a context menu by holding the stylus by a specified time duration. The user can then choose the items in the context menu after the context menu appears. If the user fails to hold the stylus long enough, nothing will happen.

The context-menu gesture by a tap-and-hold method, according to *Chew*, is different from the claimed invention in many ways.

- (A) If the contact of the stylus is short, according to *Chew*, nothing will happen. No command is carried out. In contrast, according to the claimed invention, a command assigned to the designated area will be carried out. *Chew* does not disclose or suggest that a command is assigned to a designated area if the stylus contact is brief.
- (B) If the contact of the stylus is long enough, a context menu may appear, according to *Chew*, if the parent application wants to. In contrast, in the claimed invention, a message associated with the command appears. It is respectfully submitted that the context menu 400 or 704 is not a message associated with a command assigned to a designated area. In *Chew*, the context menu 400 is generic to all the entries 302 on the contact list 300. Likewise, the context menu 704 is generic to all the options 702 on the task application list 700. The generic context menu 400 includes a set of commands 404. The user can cause a command to be carried out, according to *Chew*, by further tapping the selected command in the context menu (see col. 4, lines 27-33). Further tapping is irrelevant to the claimed invention.
- (C) After the context menu appears in response to the sufficiently long contact, according to *Chew*, the user may dismiss the context menu by tapping the stylus outside the context menu without making a selection (col.4, lines 32-34). By tapping the stylus outside the context menu, the user must lift the stylus from the screen. However, *Chew* does not specifically disclose what event will occur when the stylus is lifted after the context menu appears but the user fails to tap the stylus outside the context menu area. In contrast, according to the claimed invention, when the user lifts the physical object after message appears in response to a sufficiently long contact, the command with which the message is associated is also carried out. *Chew* does not disclose or suggest this feature.

(D) If the user drags the stylus outside the designated area, according to *Chew*, nothing will happen because the stylus move event is considered as an event not within the gesture parameter. In contrast, according to the claimed invention, the message ends if the physical object is moved out of the designated area while keeping the physical object on the screen. *Chew* does not disclose or suggest this feature.

On page 2 of the office action, the Examiner states that the electronic device is capable of carrying out a command (input function) and further capable of providing a message (context menu or tool tip) associated with the command.

It is respectfully submitted that the command assigned to a designated area, according to the claimed invention, is carried out if the contact is shorter than the predetermined time duration (step 2) or the physical object is removed from the screen after the message is displayed (step 4). *Chew* does not disclose or suggest that the “input function” is carried out if the stylus contact is shorter than the predetermined time duration. On the contrary, if the contact of the stylus or the physical object is short, according to *Chew*, nothing will happen (see point A above). Furthermore, according to the claimed invention, the message associated with the command is displayed if the contact is longer than the predetermined time duration. According to *Chew*, a context menu will appear if the contact is sufficiently long. However, the context menu is not a message associated with a command which is symbolized by an icon displayed at a designated area. The context menu is generic to the items in the contact or task application list. Moreover, the context menu is not associated with the “command (input function)” that is carried out whenever the contact is short. In *Chew*, the input function is selectable from the context menu which appears on the screen only when the contact is sufficiently long.

Since the combination of the teachings in *Chew* and *Clark* does not produce all of the claim elements in claim 1 as shown in points A to D above, it is respectfully submitted that *Chew*, in view of *Clark*, does not render the invention as claimed in claim 1 obvious.

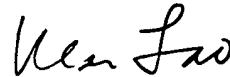
Claims 7 and 13 have all the limitations of claim 1. For the same reasons, *Chew*, in view of *Clark*, does not render the invention as claimed in claims 7 and 13 obvious.

As for the remaining claims, they are dependent from claims 1, 7 and 13 and recite features not recited in claims 1, 7 and 13. For reasons regarding claims 1, 7 and 13 above, claims 2, 3-6, 9-12 and 15 are distinguishable over the cited *Chew* and *Clark* references.

CONCLUSION

Claims 1, 3-7, 9-13 and 15 are allowable. Early allowance of all pending claims is earnestly solicited.

Respectfully submitted,



Kenneth Q. Lao
Attorney for the Applicant
Registration No. 40,061

WARE, FRESSOLA, VAN DER SLUYS
& ADOLPHSON LLP
Bradford Green, Building Five
755 Main Street, P.O. Box 224
Monroe, CT 06468
Telephone: (203) 261-1234
Facsimile: (203) 261-5676
USPTO Customer No. 004955